PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 13/00, 13/14

A1

(11) International Publication Number: WO 96/13775

(43) International Publication Date: 9 May 1996 (09.05.96)

(21) International Application Number: PCT/US95/13309

(22) International Filing Date: 13 October 1995 (13.10.95)

(30) Priority Data:
08/329,707 26 October 1994 (26.10.94) US

(71) Applicant: FLAMEPOINT, INC. [US/US]; 60 Canyon Cove Drive, Logan, UT 84321 (US).

(72) Inventor: YOSHIMOTO, Hitoshi; P.O. Box 3687, Logan, UT 84323-3687 (US).

(74) Agents: THORPE, Calvin, E. et al.; Thorpe, North & Western, 9035 South 700 East, Sandy, UT 84070 (US).

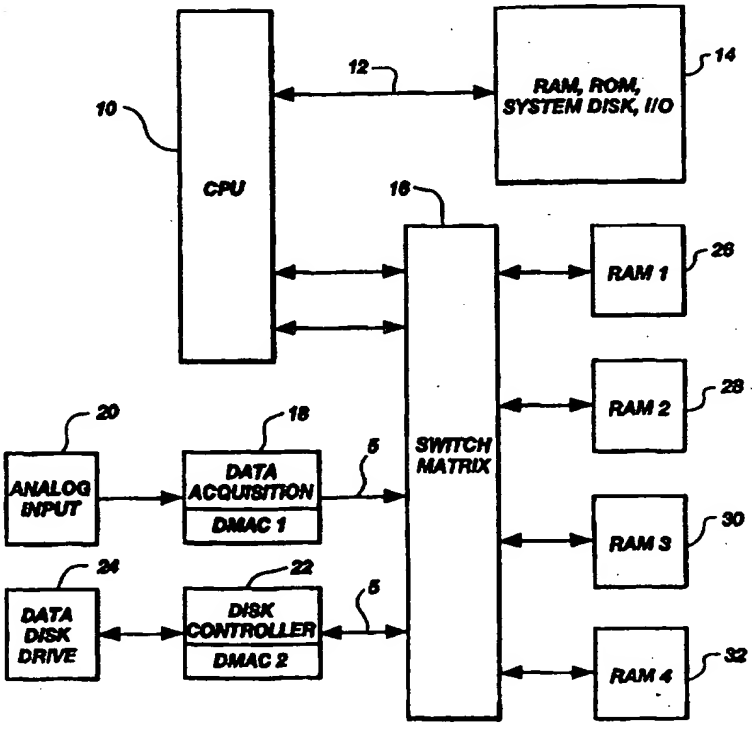
(81) Designated States: AL, AM, AT, AU, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, EE, ES, FI, GB, GE, HU, JP, KE, KG, KP, KR, KZ, LK, LR, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SI, SK, TJ, TT, UA, US, UZ, VN, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG), ARIPO patent (KE, MW, SD, SZ, UG).

Published
With international search report.

(54) Title: SIMULTANEOUS PROCESSING BY MULTIPLE COMPONENTS

(57) Abstract

A system for simultaneous processing of multiple data streams by multiple components of a computer system. The system expedites simultaneous acquisition, processing, displaying, transferring, retrieval, and storage in an arrangement with a CPU (10), I/O devices (18, 22), loopback connections and optional multiple memory banks (26, 28, 30, 32) connected to a crossbar switch matrix (CBSM) (16). A system disk (14) is separately connected to the CPU (10) on a CPU bus (12) and is isolated physically and electrically from the other components. During transfer of data to or from any of the devices connected to the CSBM (16), the CPU (10) carries out its periodic routine system overhead operations without interrupting data flow to and from other devices because the CPU (10) has a separate CPU bus (12) providing a connection to the system disk and other system components. Thus data transfer operations are continued uninterrupted and completed expeditiously.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	**GB**	United Kingdom	**MR**	Mauritania
AU	Australia	**GE**	Georgia	**MW**	Malawi
BB	Barbados	**GN**	Guinea	**NE**	Niger
BE	Belgium	**GR**	Greece	**NL**	Netherlands
BF	Burkina Faso	**HU**	Hungary	**NO**	Norway
BG	Bulgaria	**IE**	Ireland	**NZ**	New Zealand
BJ	Benin	**IT**	Italy	**PL**	Poland
BR	Brazil	**JP**	Japan	**PT**	Portugal
BY	Belarus	**KE**	Kenya	**RO**	Romania
CA	Canada	**KG**	Kyrgystan	**RU**	Russian Federation
CF	Central African Republic	**KP**	Democratic People's Republic of Korea	**SD**	Sudan
CG	Congo			**SE**	Sweden
CH	Switzerland	**KR**	Republic of Korea	**SI**	Slovenia
CI	Côte d'Ivoire	**KZ**	Kazakhstan	**SK**	Slovakia
CM	Cameroon	**LI**	Liechtenstein	**SN**	Senegal
CN	China	**LK**	Sri Lanka	**TD**	Chad
CS	Czechoslovakia	**LU**	Luxembourg	**TG**	Togo
CZ	Czech Republic	**LV**	Latvia	**TJ**	Tajikistan
DE	Germany	**MC**	Monaco	**TT**	Trinidad and Tobago
DK	Denmark	**MD**	Republic of Moldova	**UA**	Ukraine
ES	Spain	**MG**	Madagascar	**US**	United States of America
FI	Finland	**ML**	Mali	**UZ**	Uzbekistan
FR	France	**MN**	Mongolia	**VN**	Viet Nam
GA	Gabon				

SIMULTANEOUS PROCESSING BY MULTIPLE COMPONENTS

RELATED APPLICATION

This present invention is a Continuation-In-Part application of the U.S. Patent Application Serial No. 08/329,707 which was filed on October 26, 1994.

FIELD OF THE INVENTION

The present invention relates to simultaneous processing of data sets by multiple computer components of a computer system. More particularly, the invention relates to simultaneous data flow to and from computer components uninterrupted by CPU functions involving system overhead operations or the data flow to and from other components on the data bus. To this end, the present invention could be implemented using conventional bus interfaces so as to enable users switching from common-bus computers to continue to use their existing peripheral plug-in boards.

BACKGROUND OF THE INVENTION

In a typical computer system a common-bus architecture connects all components, which may include one or several central processing units (CPU), random access memory (RAM), read-only memory (ROM), input/output (I/O) devices, disk drive controllers, direct memory access controllers (DMAC), secondary bus controllers such as a small computer systems interface (SCSI) or bus bridges to other buses such as a peripheral component interconnect (PCI) or an industry standard architecture (ISA) bus. Those components may all be disposed on a single plug-in board, or they may be implemented on a plug-in board as well as a motherboard. In the later case, the plug-in board(s) and the motherboard communicate via a bus. In some cases, data is shared by multiple

CPUs using multiple port memories or I/O devices that allow the CPUs to communicate with each other. In cases other than multiple port memories, data must be accessed by various components, one component at a time, or transferred from one component to another on a common bus, resulting in time loss as data is transferred to or from one component while the other components are idle. Such speed bottlenecks are inherent in a common-bus architecture.

For example, if an I/O device is writing data to or retrieving data from a disk drive, the CPU must interrupt the data flow to access the system disk during periodic routine system "housekeeping" operations. The data stream flow is interrupted until the CPU completes housekeeping operations, thus extending the time for data transfers to and from a computer component. Regardless of the speed of the computer components, the common-bus architecture of prior art systems slows down effective data transfer speed.

This common-bus bottleneck of data flow affects all areas of data acquisition, retrieval, processing, storage, and display. The problem is particularly acute for data which must be manipulated in real-time that may require several operations to be executed simultaneously.

Examples of typical applications that will benefit from multiple simultaneous data streams are distributed networks, multi-processor computer systems, multimedia, non-destructive testing, medical imaging, interactive computer systems such as simulations, game and virtual reality systems, systems in science and industry such as those that study fast phenomena, and diagnostic testing systems. A real-time system might have to carry out the following operations: 1) a measurable quantity such as temperature, pressure, position, velocity, or motion is converted into an electrical signal, 2) the electrical signal is digitized, 3) the digitized signal is processed and then displayed real-time, and 4) the data is stored for later review.

In such prior art systems, data acquisition and storage operations must be interrupted by each other and by CPU housekeeping operations or CPU data processing. Not only is the storage of data interrupted, but in a real-time situation where data is now often sampled at high rates, data may be lost.

When the sampling rate is near the physical tolerance of components, bus interruptions become critical and some data may not be recorded. To solve this problem it is important to have uninterrupted data flow for simultaneous data acquisition and storage functions that cannot be affected by each other or by system housekeeping operations.

There are several schemes in the prior art that attempt to solve the problem of data flow interruption by trying to speed up the transfer of data. For example, the data may be buffered to await transfer on a common bus. However, when data fragments are large, as is typically the case with graphic images, buffering becomes expensive and wasteful. If the CPU requires random access to a large portion of the image for processing, the buffer will be large and therefore expensive. In addition, a large buffer increases the amount of time the common-bus will be busy with the data transfer once processing is complete.

Another prior art approach that attempted to expedite data transfer between components is disclosed by Asfour, U.S. Patent 5,182,801. This system uses a switch to alternately allocate separate memory banks in a mutually exclusive manner between two or more CPU devices, enabling each device to operate concurrently and independently on different blocks of data. Each CPU device comprises a CPU, memory, and disk connected by a standard CPU bus, which enables connection to the switch through a device port. Asfour also discloses alternate allocation of memory banks to a CPU device and a data acquisition device, enabling one set of data to be acquired in one memory bank while another set of data is simultaneously and independently processed or stored by the CPU device without buffers. When each of the devices has completed its operation, the memory banks may be uncoupled from their allocated devices and recoupled to different allocated devices, thus effectively transferring access to a set of data from one device to another without actually moving the data within the memory banks. The efficiency and performance of the system is maximized when each device takes a similar amount of time to complete its operation on the data in each memory bank.

Asfour enables nearly instantaneous access to a set of data in a memory bank by different CPU driven devices by switching coupling of the devices to the memory bank. However, the Asfour System does not address the problem of enabling data to be simultaneously processed by a CPU device while being stored or retrieved by components within the same CPU device.

Asfour also does not address the problem of CPU interruption of input/output data flow by system housekeeping operations. In the Asfour System, each CPU device shares a bus with input/output components, including the system disk resulting in a common-bus bottleneck problem for each device.

SUMMARY OF THE INVENTION

It is an object of the present invention to improve over prior art apparati and methods of simultaneous data transfer to and from computer components in a computer system.

It is a further object of the invention to provide a method and apparatus to enable execution of computer processes to occur simultaneously. Examples of such are an I/O device acquiring data and sending it to a RAM bank, an output device reading data in a RAM bank and displaying it on an output device, a disk controller reading data in a RAM bank and storing it on a storage medium, and a CPU reading data from a RAM bank to process it.

It is a further object of the invention to provide a method and apparatus for enabling the operation of real-time processes to occur independently of the other processes and components within a common computer system.

It is a further object of the invention to provide a method and apparatus for enabling a CPU to perform routine system housekeeping operations while the data flow between all other components continues uninterrupted.

It is still another object to enable plug-in boards typically used in common bus architecture computers to be plug compatible with interface ports of the new architecture implemented in the present invention.

Still yet another object of the present invention is to provide a method and apparatus to enable I/O devices to communicate with each other independent of other computer components and data flow among them, and without interruption from other components and data flow among them.

The above and other objects of the invention are realized in an apparatus and method for eliminating data flow interruption while carrying out simultaneous rapid data transfer to and from components of a computer system. In a preferred embodiment, the invention routes communications to RAM banks by various computer components through a crossbar switch matrix (CBSM) which effectively acts in place of a data bus. The CPU is also connected to its own CPU bus, through which the CPU accesses its internal RAM, ROM and system disk. All system overhead takes place on this separate CPU bus that can be physically and electrically isolated from the CBSM connections. The isolated buses and the CBSM control circuitry effectively eliminate interruption of data flow by CPU system housekeeping operations.

In an alternative embodiment, the invention routes access between various computer I/O devices through a CBSM, instead of routing I/O devices to memory. This configuration is simpler, costs less and is useful when data from an I/O device does not require processing by the CPU before going to another I/O device. In this way, the CPU and other I/O devices are still free to act independently of all other computer components without interruption of data flow even when more than two continuous data flows exist concurrently over the CBSM bus.

Connections to the CBSM is through interface ports of the CBSM whereby peripheral device plug-in boards electrically and mechanically couple via slots. It is possible to implement theinvention in such a way that no modification to the plug-in boards is required to take advantage of the increased data throughput of the system, thereby enabling backwards compatibility with pre-existing plug-in boards.

A preferred embodiment of the invention provides real-time application in which several operations are conducted simultaneously. For example, data is acquired by an I/O device accessed through a plug-in board and coupled to the CBSM through a standard plug-in board slot, said data to be stored in one RAM bank, previously acquired data in another RAM bank is independently and concurrently accessed through the CBSM by the CPU for processing, and data previously processed and stored in still another RAM bank is accessed

through the CBSM by a disk controller, also accessed through a plug-in board connected to a CBSM slot, that stores data to disk. When the CPU must inevitably conduct a system housekeeping task, the CPU communicates with the system disk over the isolated CPU bus. None of the operations conflict because the CBSM routes data flow independently.

In an alternative embodiment, data is acquired by an I/O device accessed through a plug-in board and coupled to the CBSM through a standard plug-in board slot. The CBSM routes the data to a different I/O device on either the opposite or the same side of the CBSM to enable a continuous or intermittent data stream between the I/O devices, independent of all other computer functions.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other objects, aspects, and embodiments of the present invention are described in the following specification with reference to the attached drawing figures, of which:

FIG. 1A is a block diagram of a preferred embodiment of the present invention comprising an arrangement of computer components for data acquisition, processing, and storage;

FIG. 1B is a view of a prior art common bus showing functional implementation of the bus slot connectors and controller cards.

FIG. 1C is a view of the preferred embodiment of the present invention wherein a CBSM has replaced the common bus of FIG. 1B.

FIG. 1D is a block diagram of an alternative embodiment of the present invention of FIG. 1A comprising computer I/O devices on one or both sides of the CBSM.

FIG. 1E is a block diagram of an alternative embodiment of the present invention with a modification of FIG. 1D, whereby I/O devices on the same side of the CBSM can communicate by routing communication through a loopback connection.

FIG. 2 is a block diagram of the internal structure of a CBSM comprising an arrangement of cross-bar switches;

FIG. 3 is a block diagram of the arrangement of FIG. 1 showing greater details of the connections between computer components;

FIG. 4A is a block diagram of a different preferred embodiment of the present invention comprising an arrangement of computer components for a multimedia system;

FIG. 4B is a rearrangement of the embodiment of FIG. 4A illustrating a different method of controlling the CBSM in a multimedia system;

FIG. 5A is a block diagram showing a prior art embodiment of a multiple-CPU system;

FIG. 5B is a block diagram of a preferred embodiment of a multiple-CPU system that illustrates the advantage of the present invention;

FIG. 6 is a block diagram of a different preferred embodiment of the present invention comprising an arrangement of computer components for a movie-on-demand system or digital library;

FIG. 7A is a timing diagram for the present invention that illustrates the movie-on-demand embodiment of FIG. 6; and

FIG. 7B is a timing diagram for the present invention that further illustrates the movie-on-demand system.

DETAILED DESCRIPTION

FIG. 1A illustrates a preferred embodiment of an apparatus according to the present invention comprising an arrangement of computer components for data acquisition, processing, and storage. A CPU 10 is connected through a bus 12 to a RAM, ROM, system disk, and I/O device, collectively labeled 14. The CPU 10 is also connected to two ports of a CBSM 16 or equivalent circuitry as detailed below. A data acquisition device 18, receiving analog signal input 20, is connected to the CBSM on the I/O side (shown as the left side in FIG. 1) of the CBSM, as is the CPU. A disk controller 22 controlling a disk drive 24 is also connected to the I/O side of the CBSM. Four banks of RAM 26, 28, 30, and 32 are connected to the memory side of the CBSM opposite the CPU 10, data acquisition device 18, and disk controller 22. It should be noted that analog input device 20 and disk drive 24 are generically considered as data transfer devices. As such, this invention should be

considered to include but not be limited to any data transfer device such as a data acquisition unit, a data storage unit, a display device, an audio interface unit, an I/O port, a direct memory access controller, a network interface unit, and an I/O processor, such as a bus bridge device.

This configuration of components provides for independent data streams not only in acquisition and processing, but in the recording of data to storage medium and the system overhead operations of the CPU.

A feature of the preferred embodiment that is not readily apparent from the block diagram of FIG. 1A is the physical structure of the connections between the CBSM and I/O devices. In FIG. 1A, the I/O devices 20 and 24 communicate with controllers which interface with the CBSM 12, providing low level communication protocols and physical pin connections. The I/O device controllers of FIG. 1A are the data acquisition device 18 with its accompanying DMAC, and the Disk Controller 22 with its accompanying DMAC. Each of these controllers 18 and 22 is in turn electrically coupled by a cable or some other communication medium to a specific I/O device, in this case the Analog Input 20 and the Data Disk drive 24 respectively.

Controllers such as 18 and 22 interface with the CBSM 16 in a preferred embodiment through communication ports, commonly referred to as bus slots 350, 351, 352, 353 and 354 as shown in FIG. 1C. However, "communication ports" should not be considered limited to bus slots. The term refers to any interface between the CBSM 16 and an I/O device, such as a hard-wired port on a motherboard which requires no physical slots. Nevertheless, in a preferred embodiment, the controllers are shown as the controller cards 355, 356, 357, 358 and 359 which are inserted in the aforementioned bus slots. A key feature of the present invention which provides a significant advantage to the user of a computer system based on the CBSM architecture is the implementation of the CBSM 16 bus slots 350, 351, 352, 353 and 354 as industry standard interfaces, in terms of software and/or hardware. An industry standard bus slot implementation is shown for comparison purposes in FIG. 1B, wherein slots 360, 361, 362, 363 and 364 comprise the bus slots of a common-bus architecture system wherein the same

controller cards 355, 356, 357, 358 and 359 are inserted. The only discernible difference shown is that the bus 370 of FIG. 1B is replaced by the CBSM 16 of FIG. 1C. This architectural difference could be made to be transparent to the computer application program, manifesting itself only through increased data throughput.

By using industry standard bus slots, controllers 18 and 22 can be the same controllers used in millions of common bus architecture computers. For example, the bus slots 350, 351, 352, 353 and 354 of the preferred embodiment shown in FIG. 1C support Industry Standard Architecture (ISA), Extended Industry Standard Architecture (EISA), Peripheral Component Interconnect (PCI), Microchannel, NuBus, Versa Module Europe (VME), VMEbus eXtensions for Instrumentation (VXI), FurtureBus, and other common bus slot interfaces. These bus slot structures enable existing peripheral device plug-in boards to electrically couple to the CBSM without modification to the plug-in board hardware. Being backwards compatible in bus slot interfaces eliminates what would otherwise be a significant drawback to implementation of a new computer architecture.

Throughout the remainder of this disclosure, the block diagrams of FIGs. 1D, 1E, 3, 4A, 4B and 6 will functionally describe the interconnections between components of the present invention. It is an element of the present invention that whenever these figures refer to I/O devices, a controller device with an arrow to the CBSM is functioning as an interface between the CBSM and the I/O device. It is also an element of the present invention that in a preferred embodiment, the controller device is a plug-in board inserted in a receiving slot of the CBSM, and that the receiving slot is the same as described in FIG. 1C and shown as the dotted line designated 5. Any functional arrow of FIGs. 1D, 1E, 3, 4A, 4B and 6 designated as 5 may be assumed to be implemented using the functional receiving slot structure shown in FIG. 1C.

In a typical operation, the data acquisition device 18 receives analog signals from the analog input 20. The data acquisition device 18 digitizes the signal 20 and places the digitized data in RAM bank 26. If the data is to be processed before it is stored, RAM bank 26 may be switched to communicate

with the CPU which processes the data and stores the processed data, for example, in RAM bank 28. While the CPU is handling the data in RAM banks 26 and 28, the data acquisition device 18 continues its operations using other RAM banks, such as RAM bank 30. The switching of CBSM 16 pathways is carried out by means of control logic known to those skilled in the art.

After the CPU finishes with RAM bank 28, the RAM bank switches to the disk controller 22 which records the processed data on the data disk drive 24. While the processed data in RAM bank 28 is being recorded, the data acquisition device 18 places more input data in RAM bank 30 and the data acquisition device 18 continues its operation again using RAM bank 26. At that time, the CPU 10 processes data in RAM bank 30 and places the results into RAM bank 32. After the disk controller 22 completes recording the processed data from RAM bank 28, it switches to RAM bank 32 and records the processed data therein. The process of acquisition and processing may then be repeated. The three operations of acquisition, processing, and storage can occur simultaneously due to the characteristics of the cross bar switch matrix or an equivalent circuit, as explained later.

It can be seen that the data flow to a storage device is carried out independent of acquisition and processing of other data by other devices in the present invention, and thus need not be interrupted by other processes functioning concurrently. This continuous data flow is enabled because of the separation of the I/O data disk 24, physically as well as electrically, from the system disk 14 and the CPU 10.

This arrangement has a further advantage. When the CPU 10 executes a routine system overhead operation for which it requires access to the system disk, it need not interrupt the acquisition or storage of data streams. Rather the CPU acts independent of the data flow to the other components because of its direct connection to the CPU bus 12.

FIG. 1D shows an alternate embodiment of the preferred embodiment of FIG. 1A. The modification consists of replacement of some or all of the RAM banks with I/O devices 21 and their controllers 19. A further

modification is implementation of the industry standard bus slots 5 which connect the I/O devices to the CBSM bus. This configuration provides the advantage of enabling I/O devices 20, 21 and 24 to communicate without interruption when data does not require processing by the CPU 10. This does not mean that data cannot be processed by the I/O devices, 20, 21 and 24 or their controllers 18, 19 and 22, but removes the extra steps of storing data in RAM and then moving data from RAM to another I/O device when no CPU 10 processing is required. Furthermore, the figure shows that flexibility is maintained in the system by not eliminating RAM banks from CBSM bus paths. Therefore, when data requires processing or temporary RAM storage, the RAM banks 26 and 28 are still available for this purpose. As stated above, a further configuration might be to remove all of the RAM banks. This configuration would be useful in a dedicated machine when it is known that no data will ever require CPU processing when transferring data from one I/O device to another through the CBSM bus.

FIG. 1E illustrates another alternative embodiment of the preferred embodiment of FIG. 1A. The modification consists of replacing some or all of the RAM banks with bus-like connection 23. This connection 23 provides the advantage of enabling I/O devices 20 and 24 to communicate directly and without interruption when data does not require processing by the CPU 10. In other words, this embodiment has the same advantage as FIG. 1D of allowing direct communication between any two devices attached to the device ports of the CBSM. But in addition, this embodiment allows the CPU 10 to communicate to any I/O device directly. Advantageously, this allows for faster communication between devices when blocks of data exchanged between them are significantly smaller than the RAM blocks and no data processing needs to occur between data transfer.

For example, if data from a first device must first be sent to a RAM block, and then from the RAM block to a second device, data transfer actually becomes slower using the CBSM data switching technique illustrated in FIG. 1A. Data is transferred more rapidly if RAM blocks are bypassed altogether as

illustrated in FIG. 1E where loopback connection 23 allows two data transfer devices on the same side of the CBSM to transfer data directly.

The use of a CBSM is not essential to the invention as long as a suitable substitute is used for the routing of the data streams, such as an array of gates comprising bi-directional buffers. CBSMs are known in the art as devices which enable the connection of a component on one side to any component on the other side of the CBSM, rather than connecting two components on the same side of the CBSM. An illustrative CBSM is shown in FIG. 2. This cross bar switch matrix 16 is an arrangement of six individual cross bar switches 300, 302, 304, 310, 312, and 314, such as the QS 3383 made by Quality Semiconductor. On the I/O side of the CBSM 16 are four I/O Ports 316, 318, 320, and 322. These ports connect to any I/O device such as a CPU. By controlling the path through the CBSM 16, Port 1 (316) can be connected to any one port on the memory side of the CBSM 16. The memory ports are numbered 324, 326, 328, and 330. Ports on a same side such as 316, 318, 320, and 322 can not connect to each other, nor can a single port connect to more than one port at a time on the opposite side of the matrix.

FIG. 3 shows a block diagram of the arrangement of FIG. 1, showing more details of a preferred embodiment of the present invention. The CPU 10 is connected to its components 14 along bus 12, including a DMAC 14a, ROM 14b, RAM 14c, system disk 14d and I/O device 14e. CPU 10 communicates with CBSM 16 via connections 35 and 37 between ports 36 and 38 on the CBSM and the CPU bus 12. Connections 35 and 37 should contain both address and data lines because each RAM block 26, 28, 30 and 32 needs to be accessed with both. Other control signals such as read-write and timing are also required for proper operation as is known to those skilled in the art. For simplicity, however, they have been omitted from the diagram.

Some of the address lines may be decoded and converted into chip select signals for RAM banks 26, 28, 30, 32. This process of decoding and conversion into chip select lines is the same for other address lines connected between two DMACs, 18a and 22a, and two ports, 40 and 42. These two DMACs provide address signals which are necessary to access RAM blocks

that are allocated to the corresponding I/O devices, the data acquisition 18b and disk controller 22b. Similarly, read-write and other timing and control signals are generated by the DMACs 18a and 22a, just as they are by the CPU 10.

The DMACs 18a, 22a and I/O devices 18b, 22b may need initialization, programming, and monitoring of their status. Such operations are usually done by the CPU 10, therefore data paths 70, 72 are needed. On the other hand, in order for the DMACs to operate simultaneously with the CPU as well as with each other, such paths need to be isolated. Gates 60 and 62 enable and disable this connection.

An address decoder 46, similar to SN74LS138 from Texas Instruments, is connected to the bus 12 via a few address lines 44. The address lines 44 are those needed to control access by the CPU 10 through the data and address lines 35, 37, 54, and 56. For example, the connection between bus 12 and port 1 (36) is enabled and disabled by the control line 47. The decoder 46 is implemented in such a way that accessing a certain address with different data by the CPU 10 enables or disables the connection. An alternative embodiment would only require accessing different addresses to enable and disable the connection, regardless of the data. The control line 48 works in a similar manner for address line 37 between bus 12 and port 2 (38). The control lines 50 and 52 work in a similar manner for gates 60, 62 to enable and disable the connection between bus 12 and the DMAC-I/O device pairs, 18a, 18b and 22a, 22b, respectively. These control lines should be assigned different bits in a data byte if a single address is shared among them, or mutually exclusive addresses by the decoder 46 when different addresses are used so that they can be controlled and monitored independently.

If the CPU 10 has enough processing power after carrying out application specific processing duties, the decoder 46 could have additional control lines, collectively shown as 80, so that the CPU 10 can also work as the CBSM controller. Otherwise, an independent controller would be needed for the CBSM 16. Such a controller can be constructed using a separate microcontroller and/or a set of programmable logic devices such as field-

programmable gate arrays (FGPA). The role of the controller is to monitor the activities and status of each port of the CBSM and to change connections of the CBSM. The changes are to be made in such a way that the CBSM can 1) track requests from each port to know which port is ready to be switched, or if not ready for switching, the time when it will be ready, 2) notify each port and especially the CPU 10 the status of other ports so that the CPU 10 knows the time until a particular RAM bank will be available, 3) establish connections when all ports involved in a particular pathway become available, and 4) make certain that no connection change occurs to a port before it uses all data in the RAM bank to which it is currently assigned.

It is important to note that the electrical connections between components illustrated in FIGs. 1 and 3 do not imply that an embodiment requires all components to be manufactured on a single mother-board. The present invention can be incorporated on a single board when manufactured for new systems. However, external I/O ports, a CBSM, and RAM banks could also be constructed on a separate board. This new board would then be electrically connected to the bus of an existing CPU system. This connection could be through any existing port such as a plug-in common-bus slot, parallel port, or a serial port, depending upon the application. The advantage of such an arrangement is to allow prior art systems to be upgraded instead of replaced in order to benefit from the present invention's ability to rapidly transfer data between system components.

FIG. 4A shows a block diagram of another preferred embodiment of the present invention comprising an arrangement of computer components for a multimedia system. CBSM 16 is the same cross bar switch matrix described in FIG. 2, except that it is expanded to have six ports on each side instead of four. Four I/O controllers shown as 152, 156, 162, and 166 are also comprised of a DMAC. Specifically, an image interface 152 is connected to the I/O (left) side of the CBSM, which receives image or video input 150. An audio interface 156 is connected to the left side of the CBSM for receiving or supplying audio signals 154. These interfaces 152 and 156 may provide digitizing of incoming data if necessary. An interface controller 162 is

connected to the left side of the CBSM and is connected by means of a SCSI or a P1394 "Firewire" interface 160 or suitable substitute to a data disk drive 158 or similar device. A visual display controller 166 is also connected to the left side of the CBSM and is connected to a visual display device 164 for output of visual data. A CBSM controller 200, which might illustratively be a field programmable gate array, is shown connected to the CBSM 16 and the CPU bus 170 for logical control of the switching that is coordinated with the CPU 172. A conventional digital signal processor (DSP) 148 is connected to the left side of the CBSM 16 to assist and/or substitute the processing requirements of the CPU 172.

Several new interfaces such as the P1394 have been marketed in response to the demand for multiple kinds of data transfer at higher speeds. The P1394 enables transmissions of data including audio and video signals in real-time over a single serial communication line requiring time multiplexing and demultiplexing of different types of data into a single data stream. The present invention provides an effective means to realize such operations.

In such arrangements, it is necessary to prepare certain sets of data within a limited time frame. For instance, in the case of continuous playback of motion pictures with audio, a set of still picture frames and associated audio need to be prepared several times per second, usually 20 to 30, for smooth output.

In operation, to capture the motion picture data, the image input 150 is received from the image interface 152 and placed into RAM bank 126. Audio input 154 is sent to the audio interface 156, and then passed through the CBSM 16 and placed into RAM bank 128. The DSP 148 performs necessary data processing such as compression of the image data and places the result in RAM bank 130. The image and audio data in RAM banks 128 and 130 are then sent to the data disk 158 through the P1394 interface. The CBSM controller 200 will either act independently to enable data flow through the CBSM 16, or the CPU 172 will send control commands over control line 210 to the CBSM controller 200, or directly to the CBSM 16 over optional control line 176. The Image and Audio Interfaces 152 and 156, and the Interface and

Display Controllers 162 and 166, will communicate with the CBSM controller 200 over I/O channel control lines 104 in order to control data flow through the CBSM 16 to the RAM blocks 126, 128, 130, 132, 134, and 136.

An example of operation would be audio and video data from RAM banks 126 and 128 being compressed and stored in RAM bank 130. While RAM bank 130 is read to disk 158, the next frame of image and audio data is independently and simultaneously stored in RAM banks 132 and 134. The image is compressed and placed in RAM bank 136. That data is then sent to disk 158, and the process repeats.

To play back the motion picture stored on disk 158, the image and corresponding audio data are read from the data disk 158. The image data is placed into RAM bank 126 and the audio data is placed into RAM bank 128. The DSP 148 decompresses the image data and places the result in RAM bank 130. The decompressed image data in RAM bank 130 is then sent to the visual display controller 166 for display, and the audio data is sent to the audio interface 156 for play. While this data is being sent, image and audio data are again read from the data disk 158 and placed into RAM banks 132 and 134, respectively. The image data is decompressed and placed into RAM bank 136. The data in bank 136 is then output and the process repeats.

In this embodiment, the normal operations of the CPU can be carried out separately from the data processing operations, so that there is no interruption of the data flow of audio and video images.

FIG. 4B is also an embodiment of a multimedia system, but differs from FIG. 4A in that the CPU 172 controls the CBSM 16 through the addition of a decoder 204, a gate enabling control signal 208, and gates 212. These components replace the independently operating CBSM controller 200 of FIG. 4A. Operation of the system is similar to FIG. 4A, except that now the CPU 172 directly controls the enabling of pathways through the CBSM 16. This alternative arrangement is practical for applications where the CPU 172 is not heavily burdened, and can devote time to controlling RAM access by I/O devices. This configuration may also be less costly because the CBSM controller 200 of FIG. 4A could be costly for some applications.

FIG. 5A is a block diagram of a prior art arrangement of processors in a multiple CPU system. As shown, each CPU 220, 221, 222 has cache memory 223, 224, 225. A CPU processes data in its cache as long as needed data is found in the cache. Caching allows a CPU to avoid accessing the bus 228 to retrieve data from main memory 226 whenever data is needed, freeing bus time for other components. However, coherency must be maintained between each cache and main memory 226. Thus, CPU time as well as the bus time is lost when main memory 226 must be updated with cache contents. In addition, when cache contents do not often contain data needed by the CPU (i.e. a low cache hit), frequent updates of cache memory 223, 224, 225 results in bus contention, severely limiting efficiency of the system.

FIG. 5B is an alternative to the cache system of FIG. 5A. Instead of having a cache, the main memory 234 is divided into smaller RAM banks 235, 236, 237, 238. Each bank is allocated to a CPU 230, 231, 232 or I/O device by the CBSM 233. A CPU uses the allocated RAM block in place of cache memory. However, because this "cache" is actually part of main memory 234, there is no need to update it as in the cache system. When a CPU or I/O device needs to access different data in another RAM block, the CBSM 233 changes pathways enabling access to the desired block of main memory 234. When more than one CPU or I/O device requests access to the same RAM block, arbitration between devices will be handled at CBSM 233 switching speed instead of the slower cache data transfer speed of prior art. In such a circumstance, the operating system would need modification to accommodate the new circuitry of the present invention.

FIG. 6 shows a block diagram of a movie-on-demand system. The system is comprised of m Movie Disks, shown here as 380, 382 up to Movie Disk m 400. Movie Disks are randomly accessible digital storage devices such as hard disks or CD-ROM drives. The movie data is digitized and possibly in a compressed form. Movie Disks are connected to the I/O (left) side of the CBSM 16 through which data is transferred to RAM blocks. Also on the left side of the CBSM are n Viewers, shown here as 510, 512 up to Viewer n 600. On the right side of the CBSM there must be at least 2n RAM blocks. This

requirement is made clear in FIG. 7A and 7B. The CBSM 16 is controlled by CBSM command controller 700 which allocates RAM blocks as needed by Viewers. The CBSM controller 700 may be implemented with a micro-controller, mainframe or supercomputer, depending upon the sizes of m and n.

In this embodiment, Viewer 1 (510) will send a command signal to the command controller 700 requesting access to a movie. For example, Movie Disk 1 (380) has the requested movie that the viewer wishes to view, fast-forward, rewind, or view in an accelerated or slow-motion format. The viewer requested portion of the movie is sent through the CBSM 16 pathway created by the command controller 700 and stored in a RAM bank. When the RAM bank is filled, the command controller routes a CBSM 16 pathway to Viewer 1 (510). While Viewer 1 (510) is occupied viewing the segment of movie in RAM bank 1 (184), the next movie segment is transferred to a different RAM bank, such as RAM bank 2 (186). In this way, there is no interruption of Viewer 1's (510) movie when Viewer 1 (510) has completed watching the movie segment in RAM bank 1 (184). A sufficiently large RAM bank should be able to eliminate data gaps which cause pauses if the system has sufficient time to load the next portion of the movie before the viewer completes the current segment.

The present invention also provides the advantage of allowing simultaneous access to segments of the same movie to different viewers because of rapid transfer capabilities. If Viewer 2 (512) requests viewing Movie Disk 1 (380) while Viewer 1 (510) is also viewing the movie, two sets of RAM banks will be allocated to each viewer: RAM banks 1 and 2 (184, 186) for Viewer 1 (510), and RAM banks 3 and 4 (188, 190) for Viewer 2 (512). An illustrative example of the timing is shown in FIG. 7A. As shown, the movie disk transfer time 810 to the RAM banks must be at least twice as fast as the rate at which movies are viewed 820. Such a transfer rate may be achieved by a fast data transfer speed, or by movie compression. Movie compression would allow a large portion of a movie to be transferred to RAM in a short amount of time 810 with respect to the viewing time of the movie to

be viewed 820. The decompression is assumed to be done at the Viewer site. A combination of compression and faster transfer rates would also be possible.

As shown in FIG. 7A, Viewer 1 (510) views a segment of Movie Disk 1 (380) stored in RAM bank 2 (186) in a previous transfer, while Viewer 2 (512) views another segment of the same movie, stored in RAM bank 4 (190) in a previous transfer. During time slot 1 (810), Movie Disk 1 (380) is storing the next movie segment to be viewed by Viewer 1 (510) in RAM bank 1 (184), and in time slot 2 (830), stores the next movie segment to be viewed by Viewer 2 (512) in RAM bank 3 (188). During the next two time slots 3 and 4 (840), the CBSM 16 pathways are changed to allow Viewer 1 (510) to view the movie segment in RAM bank 1 (184), and Viewer 2 (512) views the movie segment in RAM bank 3 (188). Meanwhile, the next movie segments are loaded into RAM bank 2 (186) and bank 4 (190) for Viewer's 1 (510) and 2 (512) respectively.

By repeating the process, two viewers can access the same movie stored in the same disk simultaneously without interfering with each other. With sufficiently fast transfer rates, the present embodiment eliminates fluctuations in transfer that would causes pauses in movies being viewed because of mechanical movement within storage devices.

FIG. 7B illustrates the changes necessary in the timing scheme of FIG. 7A to enable viewing by up to n viewers of the same movie disk, where n is an arbitrary and possibly large number. The RAM banks would need to be of sufficient size, and transfer rates of sufficient speed, such that the CBSM controller would be able to transfer the subsequent segment of movie to be viewed by each viewer to the appropriate RAM bank before the current segment has been viewed. For example, while viewers 1 (510), 2 (512), ..., n (600) are watching Movie Disk 1 (380) stored in RAM banks 2, 4, ..., 2n (186, 190, ..., 198), the CBSM controller must store the next movie segments of Movie Disk 1 to be viewed in RAM banks 1, 3, ..., 2n-1 (184, 188, ..., 196) respectively. However, some reduction in the number of RAM banks is possible. Such a reduction is attained in a manner similar to the way that the total number of communication lines are reduced in network traffic, through

usage statistics. For example, a viewer needs less data to be transferred when viewing at slow motion or when paused. Statistics can indicate how often viewers need full speed data transfer, thus helping to reduce the total number of RAM banks, and thus reducing the overall cost of the entire system.

The digital library system will have a similar structure. Movie disks are replaced by library documentation disks that contain library documentation for review in digital form. The viewers would then be subscriber's to the repository. In addition, the chances are also greater that the viewers in a digital library will be reviewing still images, further reducing the total number of RAM banks required.

Other applications of this invention will be apparent to those skilled in the art, and are intended to be part of the general disclosure provided herein. The examples provided are merely exemplary of the principles, methodology and apparatus representing the subject invention. Accordingly, the specific embodiments and procedures are not to be considered as limiting with respect to the actual invention as defined by the following claims.

CLAIMS

What is claimed is:

1. Apparatus for uninterrupted data transfer to and from components in a computer system, comprising:

a switch means for selectively coupling to said components and having a plurality of ports;

a central processing unit connected to a first switch means port;

a first data transfer device connected to a second switch means port independent of the central processing unit; and

at least one memory segment being independently connected to a third switch means port.

2. The apparatus for uninterrupted data transfer as defined in Claim 1 wherein the switch means has a first side and a second side with at least one port on the first side and at least one port on the second side.

3. The apparatus for uninterrupted data transfer as defined in Claim 2 wherein the CPU and the first data transfer device are both on the first side of the switch means and the at least one memory segment is on the second side of the switch means.

4. The apparatus for uninterrupted data transfer as defined in Claim 2 wherein at least one port of the at least one port on the first side and the at least one port on the second side of the switch means is an industry standard bus slot.

5. The apparatus for uninterrupted data transfer as defined in Claim 3 wherein the apparatus further comprises a second data transfer device on the second side of the switch means.

6. The apparatus for uninterrupted data transfer as defined in Claim 3 wherein the apparatus further comprises

a second data transfer device on the first side of the switch means; and

a loopback connection coupled between two switch means ports on the second side such that the first data transfer device and the second data transfer device can communicate through the switch means independent of the CPU or other computer components.

7. The apparatus for uninterrupted data transfer as defined in Claim 3 wherein the apparatus further comprises

a second data transfer device on the first side of the switch means; and

a loopback connection on the second side of the switch means, being coupled between any two devices on the first side of the switch means so as to communicate through the switch means independent of other computer components.

8. Apparatus for uninterrupted data transfer to and from components in a computer system, comprising:

a switch means for selectively coupling to said components having a plurality of ports;

a central processing unit connected to a first switch means port; and

a first data transfer device connected to a second switch means port independent of the central processing unit.

9. The apparatus for uninterrupted data transfer as defined in Claim 8 wherein the switch means has a first side and a second side with at least one port on the first side and at least one port on the second side.

10. The apparatus for uninterrupted data transfer as defined in Claim 9 wherein the CPU and the first data transfer device are both on the first side of the switch means.

11. The apparatus for uninterrupted data transfer as defined in Claim 9 wherein at least one port of the at least one port on the first side and the at

least one port on the second side of the switch means is an industry standard bus slot.

12. The apparatus for uninterrupted data transfer as defined in Claim 10 wherein the apparatus further comprises a second data transfer device on the second side of the switch means.

13. The apparatus for uninterrupted data transfer as defined in Claim 10 wherein the apparatus further comprises

a second data transfer device on the first side of the switch means; and

a loopback connection coupled between two switch means ports on the second side such that the first data transfer device and the second data transfer device can communicate through the switch means independent of the CPU or other computer components.

14. The apparatus for uninterrupted data transfer as defined in Claim 10 wherein the apparatus further comprises:

a second data transfer device on the first side of the switch means; and

a loopback connection on the second side of the switch means, being coupled between any two devices on the first side of the switch means so as to communicate through the switch means independent of other computer components.

15. The apparatus for uninterrupted data transfer to and from components in a computer system as defined in claims 1 or 9, wherein the apparatus further comprises a central storage means connected directly to the central processing unit independent of connection to the other components.

16. The apparatus for uninterrupted data transfer to and from components in a computer system as defined in Claims 4 or 11, wherein the industry standard bus slot is selected from the group of bus connectors consisting of Industry

Standard Architecture (ISA), Extended Industry Standard Architecture (EISA), Peripheral Component Interconnect (PCI), Microchannel, NuBus, Versa Module Europe (VME), VMEbus eXtensions for Instrumentation (VXI) and FurtureBus.

17. The apparatus of Claims 1 or 8, wherein the central processing unit is connected to the central storage means by a dedicated CPU bus isolated from the other components.

18. The apparatus of Claims 1 or 8, wherein the switch means further comprises a cross bar switch matrix.

19. The apparatus of Claim 15, wherein the central storage means further comprises a system disk.

20. The apparatus of Claims 5 or 12, wherein the first and second data transfer devices are selected from the group consisting of a data acquisition unit, a data storage unit, a display device, an audio interface unit, an I/O port, a direct memory access controller, a network interface unit, and an I/O processor, such as a bus bridge device, connected to a plurality of I/O buses.

21. The apparatus of Claims 1 or 8, wherein the central processing unit is in communication with the central storage means at a remote location through a network interface device coupled to an information network.

22. A method of uninterrupted data transfer to and from components in a computer system, wherein said system has a switch device connected between computer memory and a plurality of components of the computer system that includes a central processing unit, said method comprising:

transferring data between a first component and a second component of said plurality of components via said switch device independent of said central processing unit, and

simultaneously transferring data to or from said central processing unit for data processing or storage.

23. The method of Claim 22, further including the step of simultaneously transferring data between a third component and a fourth component of said computer components independent of said central processing unit.

24. The method of Claim 22, wherein the step of transferring data to or from the central processing unit further comprises transferring data to or from a central storage unit isolated from said switch device.

25. The method of Claim 22 wherein the step of transferring data to or from the central processing unit further comprises the step of transferring data to or from a central processing unit to a system disk storage unit isolated from said switch device.

26. The method of Claim 22 wherein the step of transferring data between at least one of said components and a first portion of said memory further comprises the transfer of data from a data acquisition unit to a data storage unit.

27. An apparatus for uninterrupted data transfer between a computer system, a peripheral system, and components within the peripheral system that is electrically connected to the computer system, comprising:

1) a data transfer means coupled between said computer system and said peripheral system;

2) a central processing unit of the computer system coupled to the data transfer means;

3) a data transfer device of the computer system coupled to the data transfer means;

4) a plurality of memory segments of the computer system each coupled to the data transfer means;

5) a central storage means of the computer system coupled to the central processing unit through the data transfer means;

6) the peripheral system electrically coupled to the data transfer means through a switch means, said switch means enabling rapid data transfer between components within the peripheral system, and between components of the peripheral system and the computer system, said peripheral system comprising:

6a) a switch means for selectively coupling to components of the peripheral system and the computer system, said switch means having ports on a first side and a second side, wherein said first and/or second side ports are industry standard bus slots;

6b) a first data transfer device having an industry standard plug-in board interface for connecting to a said first side port bus slot of the switch means independent of other peripheral system or computer system components; and

6c) a second data transfer device having an industry standard plug-in board interface for connecting to a said first or second side port bus slot of the switch means independent of other peripheral system or computer system components.

28. The apparatus of Claim 27 wherein the switch means further comprises a cross bar switch matrix.

29. The apparatus of Claim 27 wherein the central storage means further comprises a system disk.

30. The apparatus of Claim 27, wherein the data transfer device is selected from the group consisting of a data acquisition unit, a data storage unit, a display device, an audio interface unit, an I/O port, a direct memory access controller, a networl interface unit and an I/O processor connected to a plurality of I/O buses.

31. The apparatus of Claim 27, wherein the peripheral system is selected from the group consisting of an internal slotted plug-in device that connects to a computer common-bus system slot, an external device that connects to a parallel port of a computer system, and an external device that connects to a serial port of a computer system.

32. The apparatus of Claim 27 wherein the data transfer means further comprises a common-bus standard.

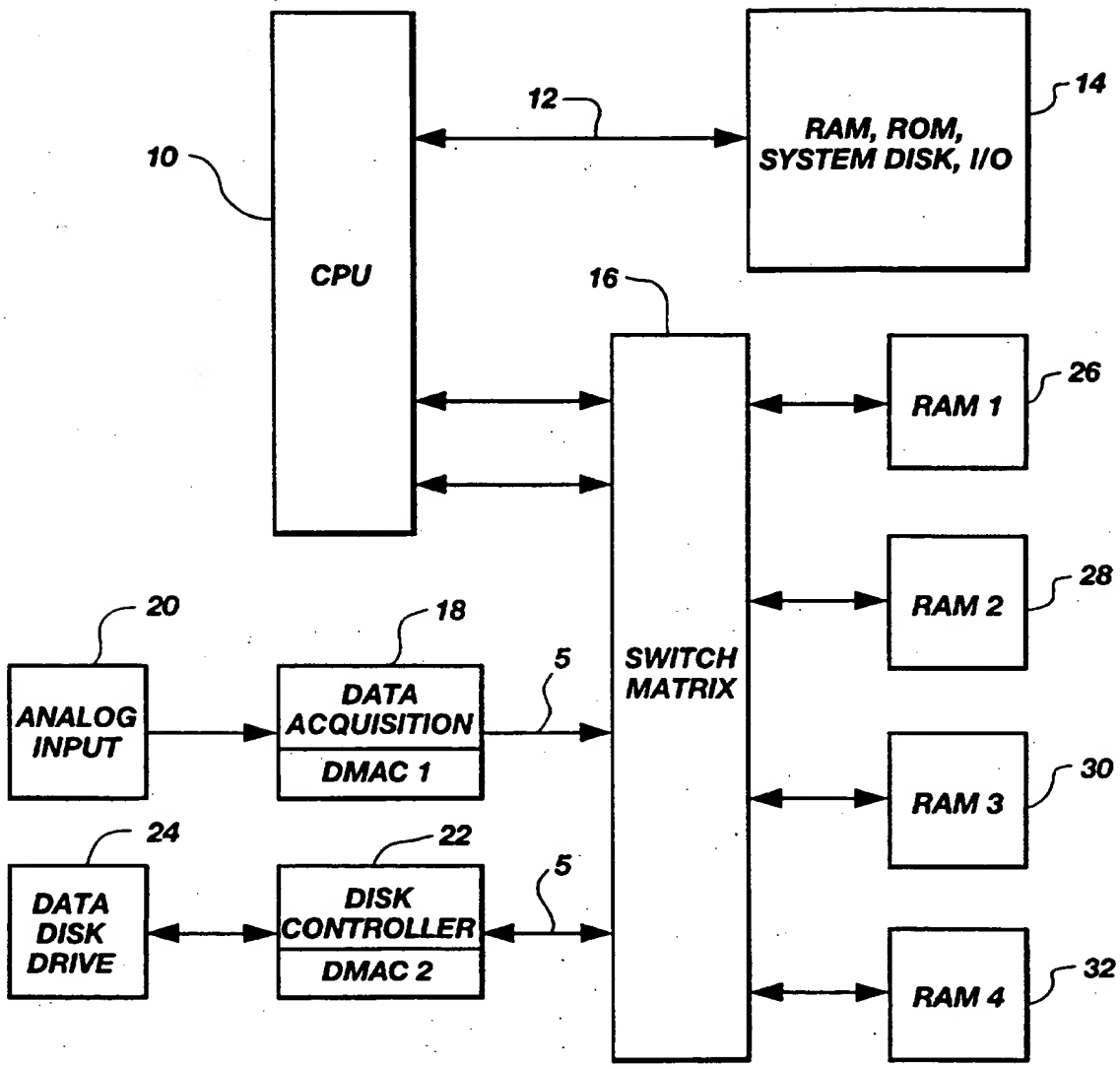

10
CPU
12
14
RAM, ROM, SYSTEM DISK, I/O
16
SWITCH MATRIX
RAM 1
26
RAM 2
28
RAM 3
30
RAM 4
32
20
ANALOG INPUT
18
DATA ACQUISITION
DMAC 1
5
24
DATA DISK DRIVE
22
DISK CONTROLLER
DMAC 2
5

Fig. 1A

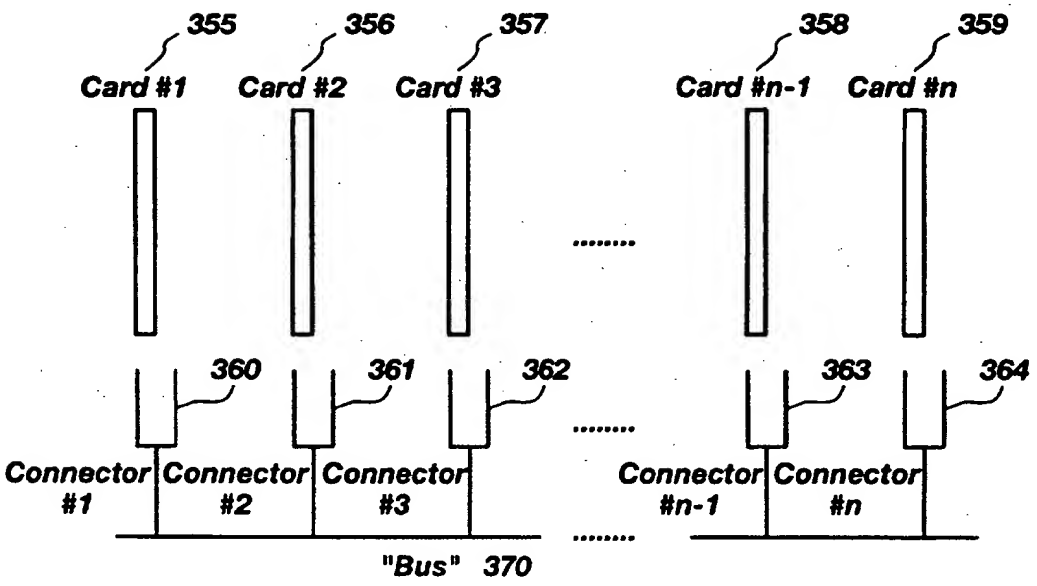

355
356
357
358
359
Card #1
Card #2
Card #3
Card #n-1
Card #n
360
361
362
363
364
Connector #1
Connector #2
Connector #3
Connector #n-1
Connector #n
"Bus" 370

Fig. 1B

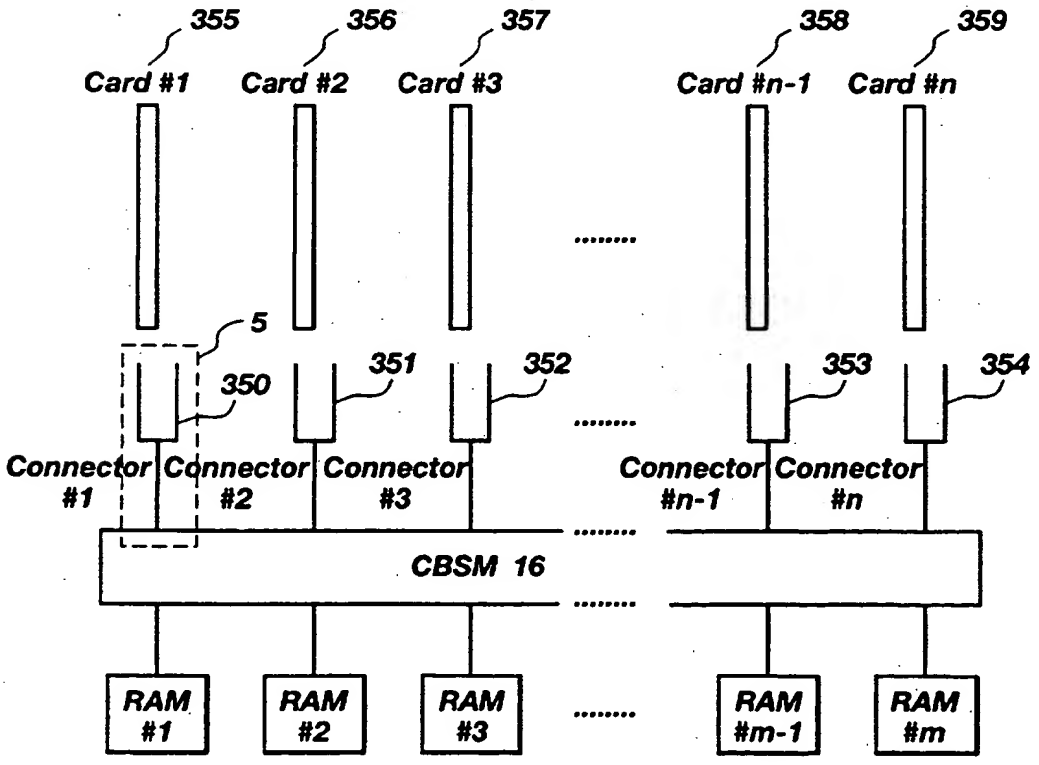

355
356
357
358
359
Card #1
Card #2
Card #3
Card #n-1
Card #n
5
350
351
352
353
354
Connector #1
Connector #2
Connector #3
Connector #n-1
Connector #n
CBSM 16
RAM #1
RAM #2
RAM #3
RAM #m-1
RAM #m

Fig. 1C

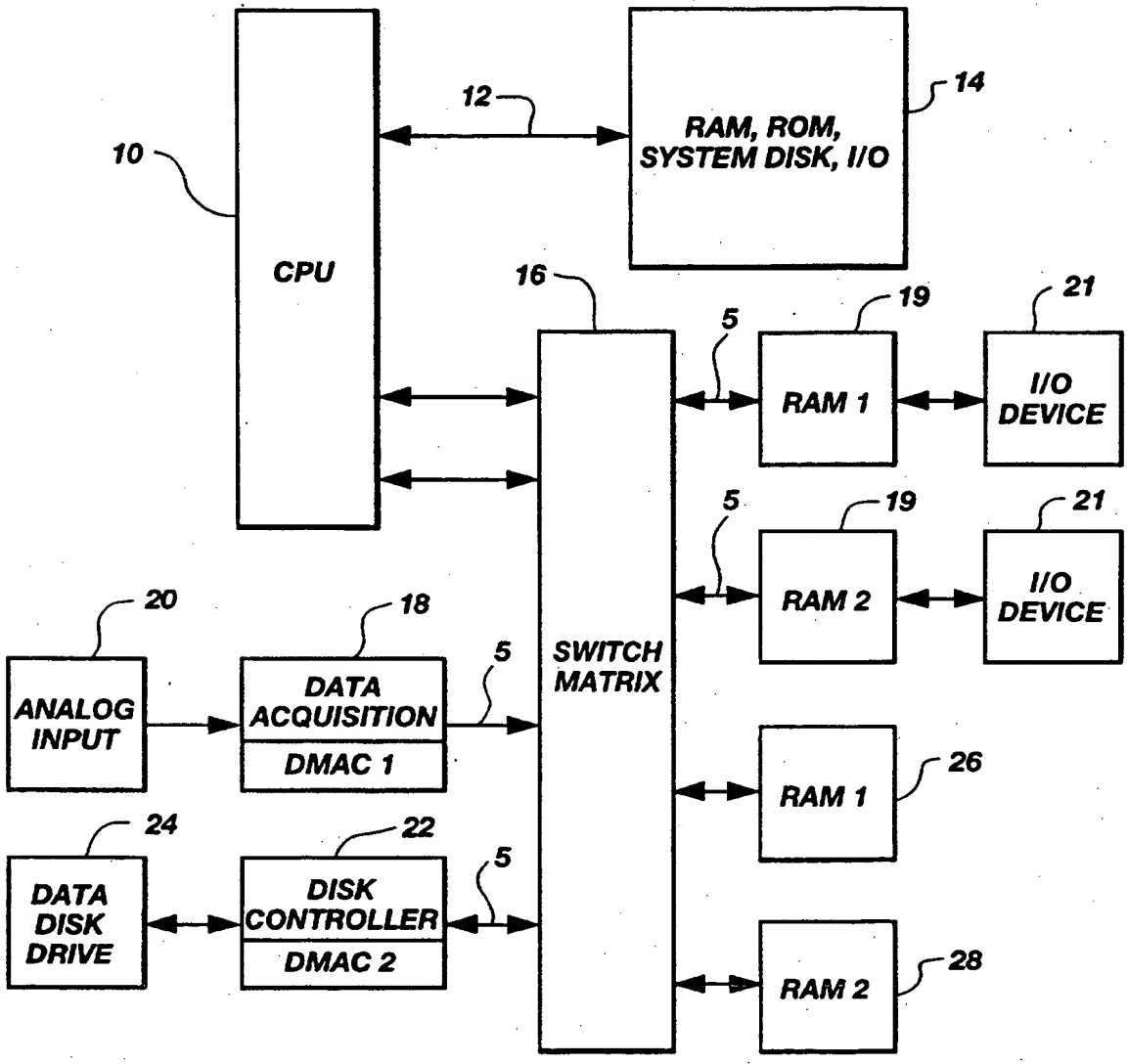

CPU
10
12
RAM, ROM, SYSTEM DISK, I/O
14
16
SWITCH MATRIX
5
RAM 1
19
I/O DEVICE
21
RAM 2
ANALOG INPUT
20
DATA ACQUISITION
DMAC 1
18
DATA DISK DRIVE
24
DISK CONTROLLER
DMAC 2
22
26
28

Fig. 1D

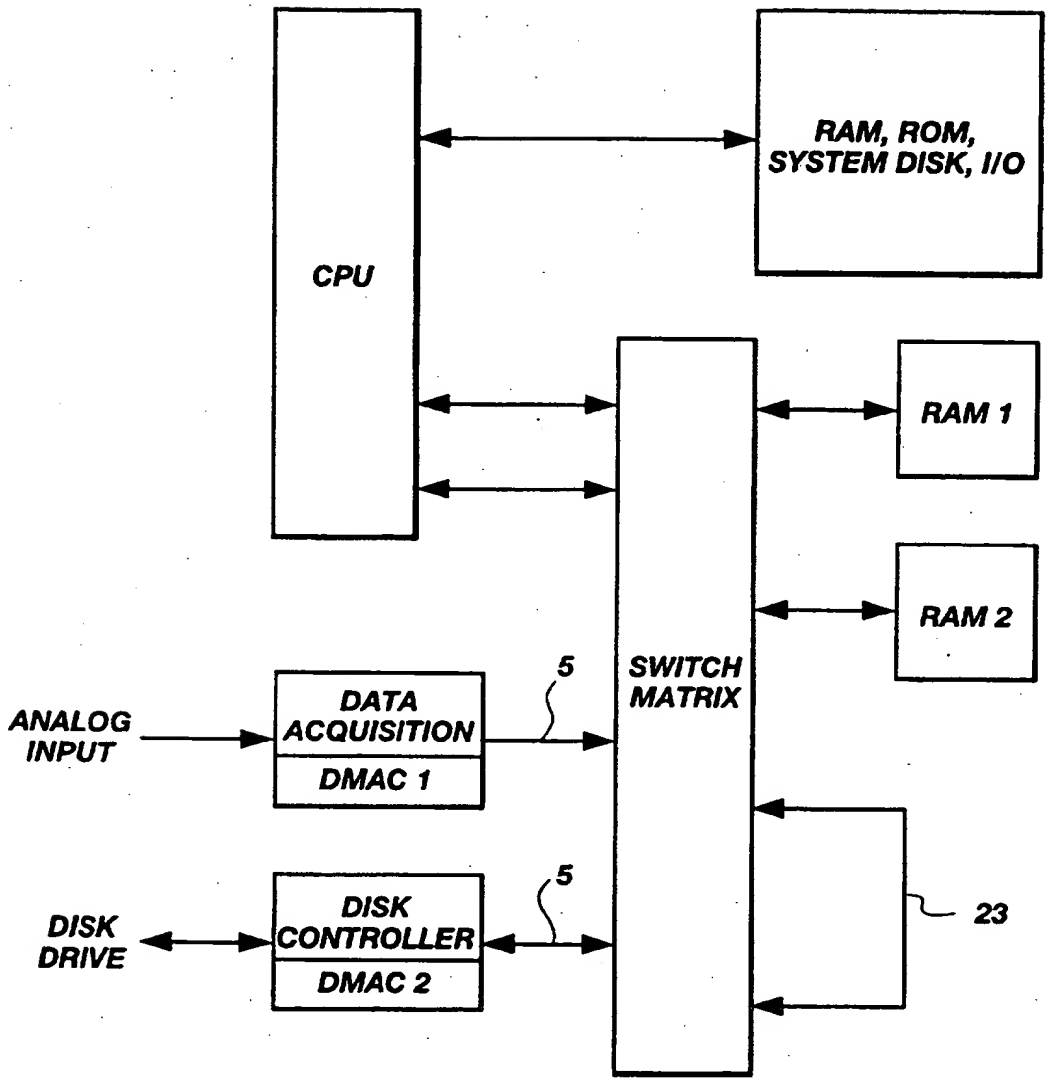

CPU
RAM, ROM, SYSTEM DISK, I/O
RAM 1
RAM 2
SWITCH MATRIX
5
ANALOG INPUT
DATA ACQUISITION
DMAC 1
5
DISK DRIVE
DISK CONTROLLER
DMAC 2
23

Fig. 1E

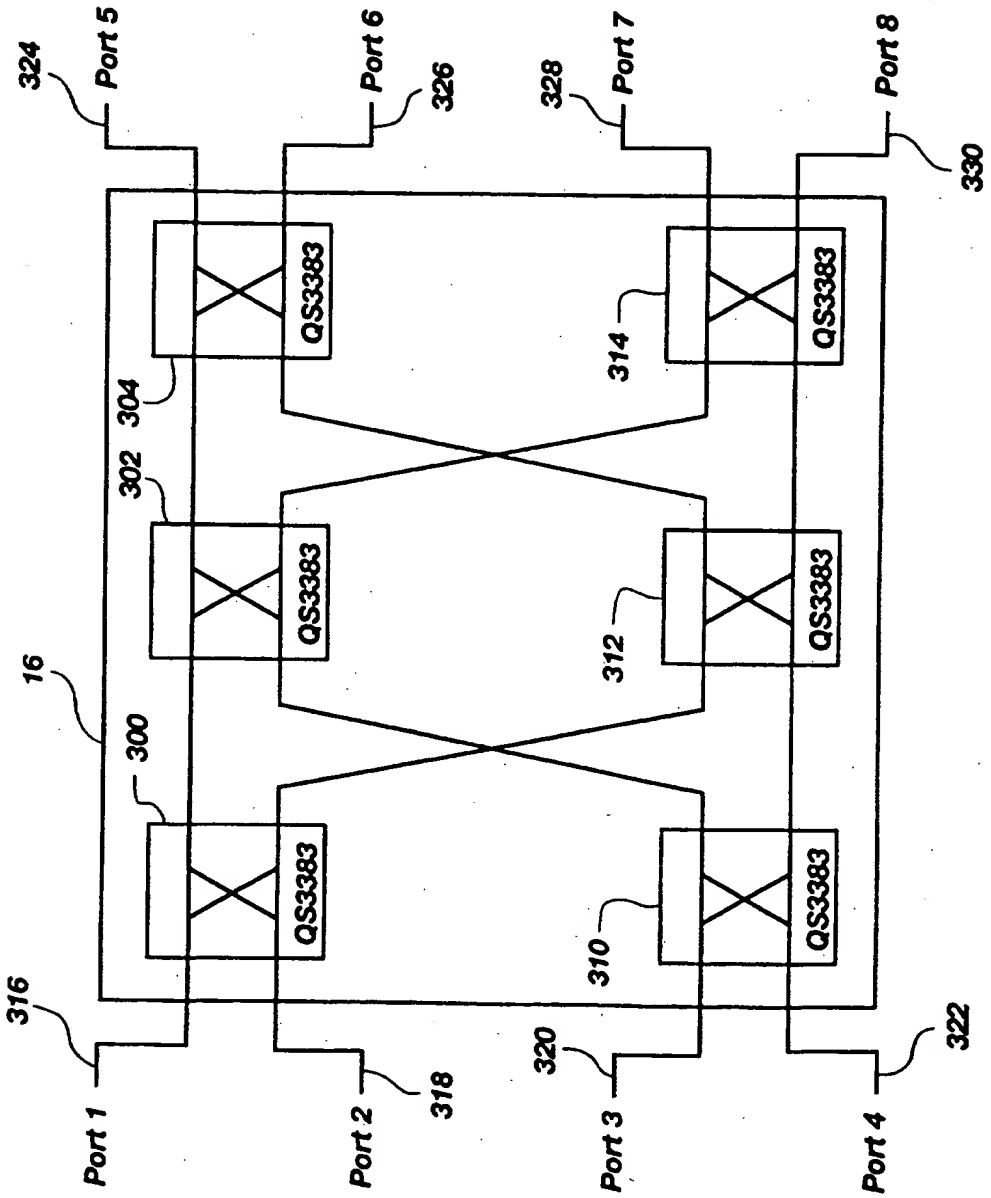

Port 5
Port 6
Port 7
Port 8
Port 1
Port 2
Port 3
Port 4
QS3383
QS3383
QS3383
QS3383
QS3383
QS3383
QS3383
QS3383
16
300
302
304
310
312
314
316
318
320
322
324
326
328
330

Fig. 2

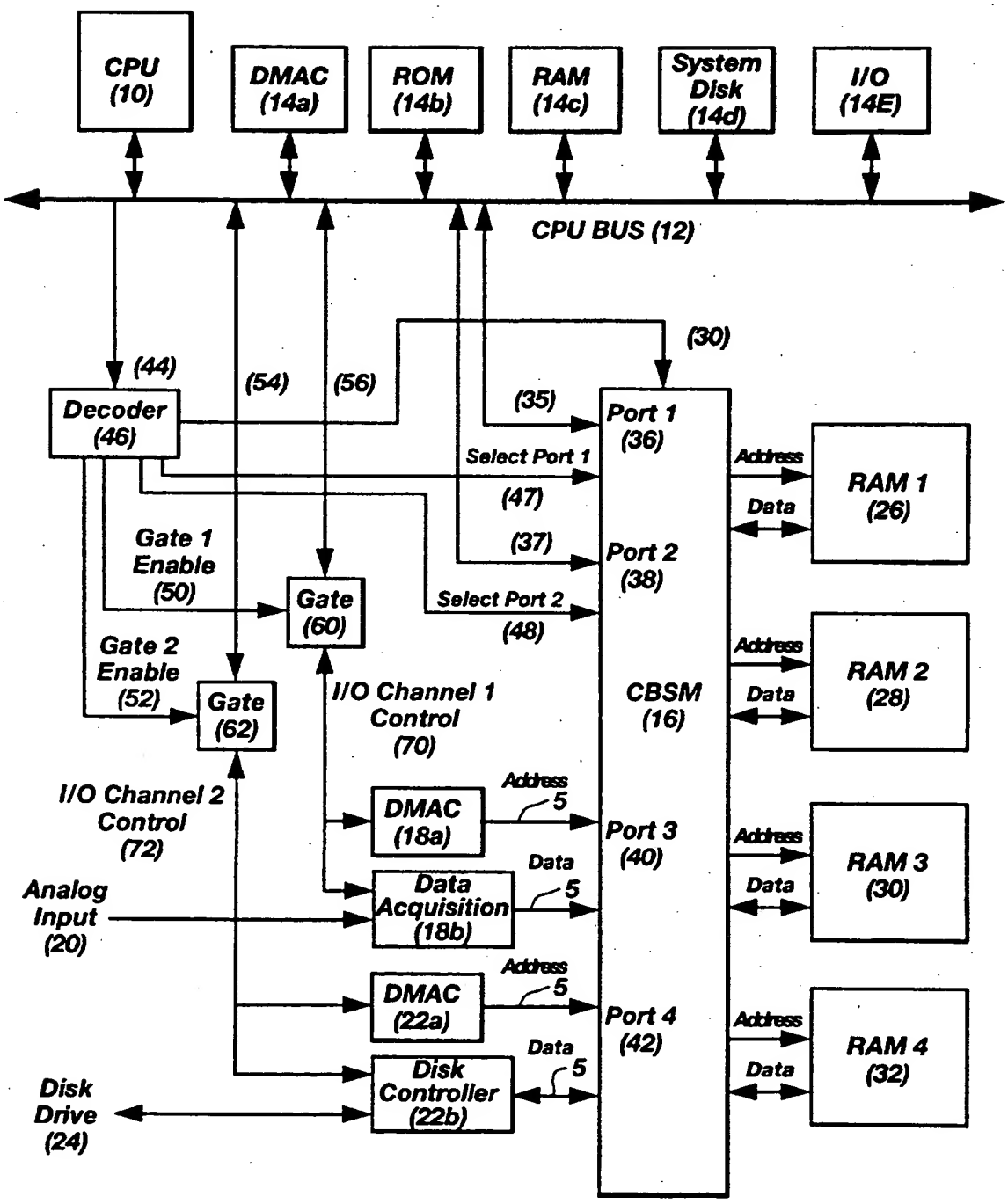

CPU (10)
DMAC (14a)
ROM (14b)
RAM (14c)
System Disk (14d)
I/O (14E)
CPU BUS (12)
(30)
(44)
(54)
(56)
(35)
Decoder (46)
Port 1 (36)
Select Port 1
(47)
Address
Data
RAM 1 (26)
(37)
Port 2 (38)
Gate 1 Enable (50)
Gate (60)
Select Port 2
(48)
Address
Data
RAM 2 (28)
Gate 2 Enable (52)
Gate (62)
I/O Channel 1 Control (70)
CBSM (16)
I/O Channel 2 Control (72)
Address
5
DMAC (18a)
Port 3 (40)
Address
Data
RAM 3 (30)
Data
5
Analog Input (20)
Data Acquisition (18b)
Address
5
DMAC (22a)
Port 4 (42)
Address
Data
RAM 4 (32)
Data
5
Disk Drive (24)
Disk Controller (22b)

Fig. 3

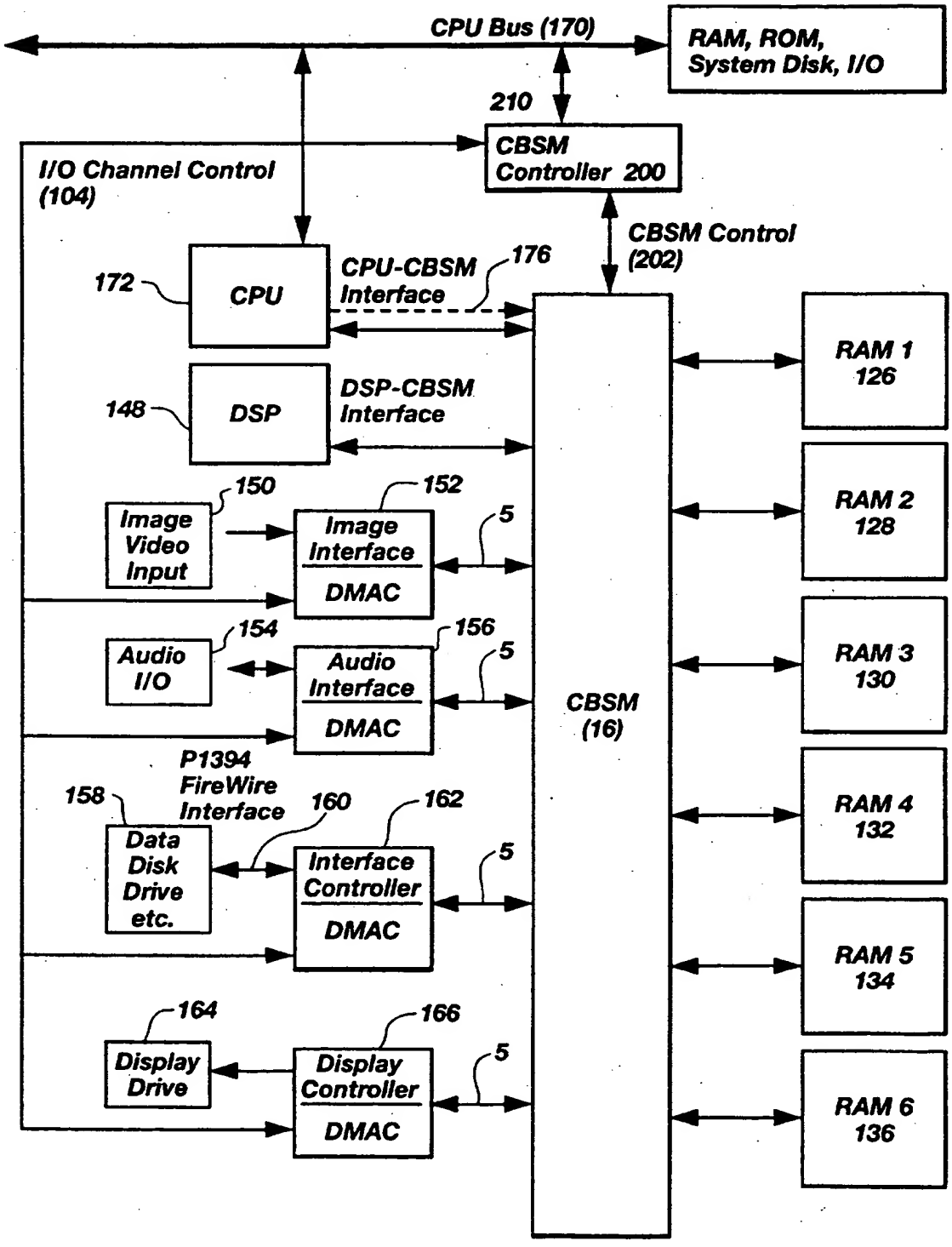

CPU Bus (170)
RAM, ROM, System Disk, I/O
210
CBSM Controller 200
I/O Channel Control (104)
CBSM Control (202)
172
CPU
CPU-CBSM Interface
176
148
DSP
DSP-CBSM Interface
150
Image Video Input
152
Image Interface
DMAC
5
154
Audio I/O
156
Audio Interface
DMAC
5
CBSM (16)
P1394 FireWire Interface
158
Data Disk Drive etc.
160
162
Interface Controller
DMAC
5
164
Display Drive
166
Display Controller
DMAC
5
RAM 1 126
RAM 2 128
RAM 3 130
RAM 4 132
RAM 5 134
RAM 6 136

Fig. 4A

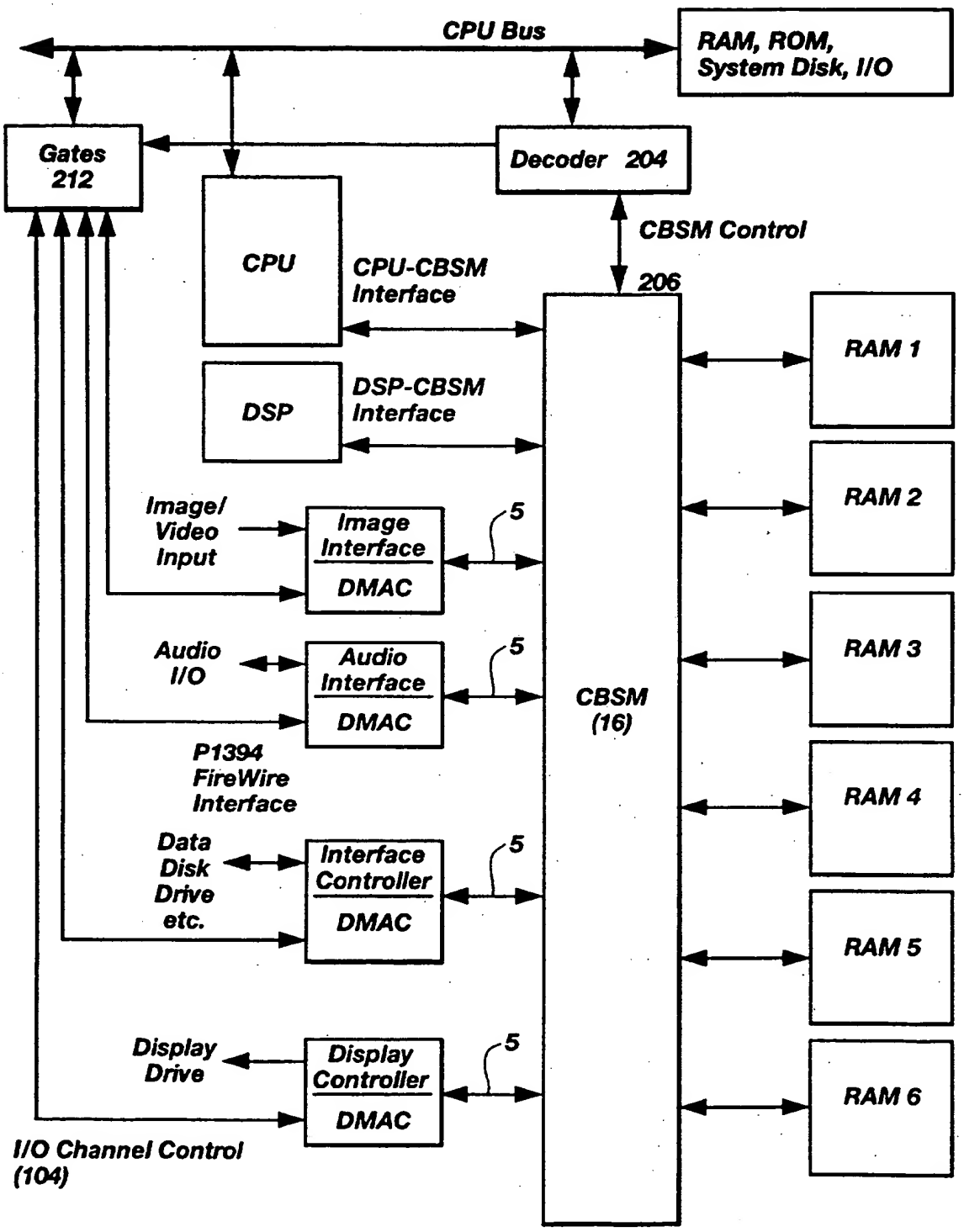

CPU Bus
RAM, ROM, System Disk, I/O
Gates 212
Decoder 204
CPU
CPU-CBSM Interface
CBSM Control
206
DSP
DSP-CBSM Interface
Image/ Video Input
Image Interface
DMAC
5
Audio I/O
Audio Interface
DMAC
5
P1394 FireWire Interface
Data Disk Drive etc.
Interface Controller
DMAC
5
Display Drive
Display Controller
DMAC
5
I/O Channel Control (104)
CBSM (16)
RAM 1
RAM 2
RAM 3
RAM 4
RAM 5
RAM 6

Fig. 4B

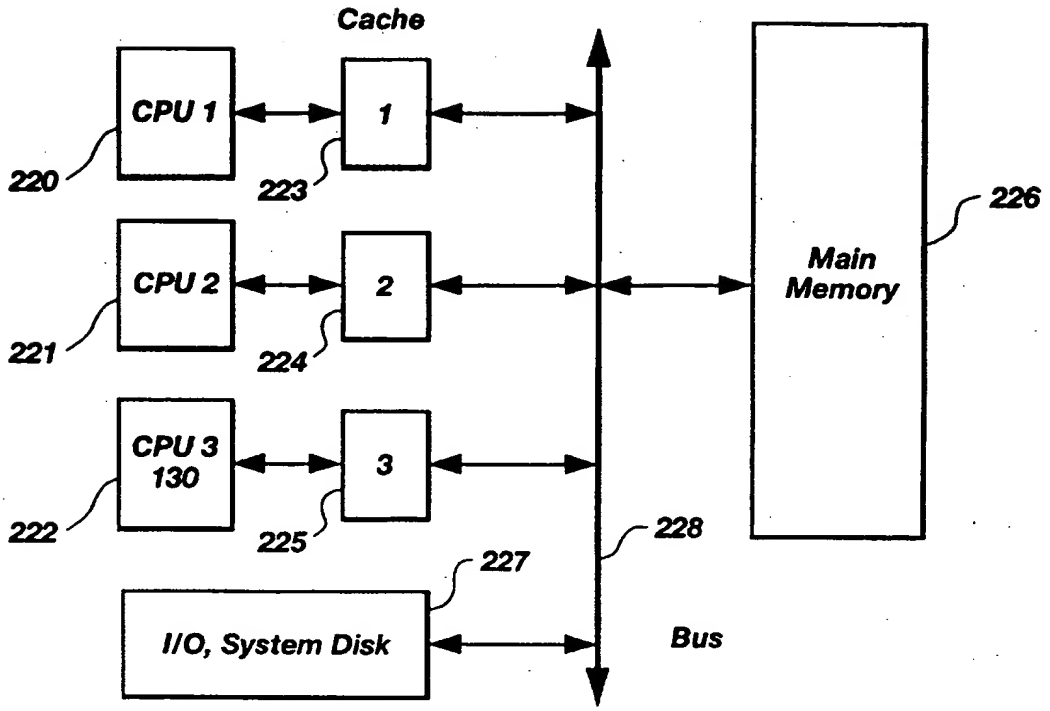


Fig. 5A
(PRIOR ART)

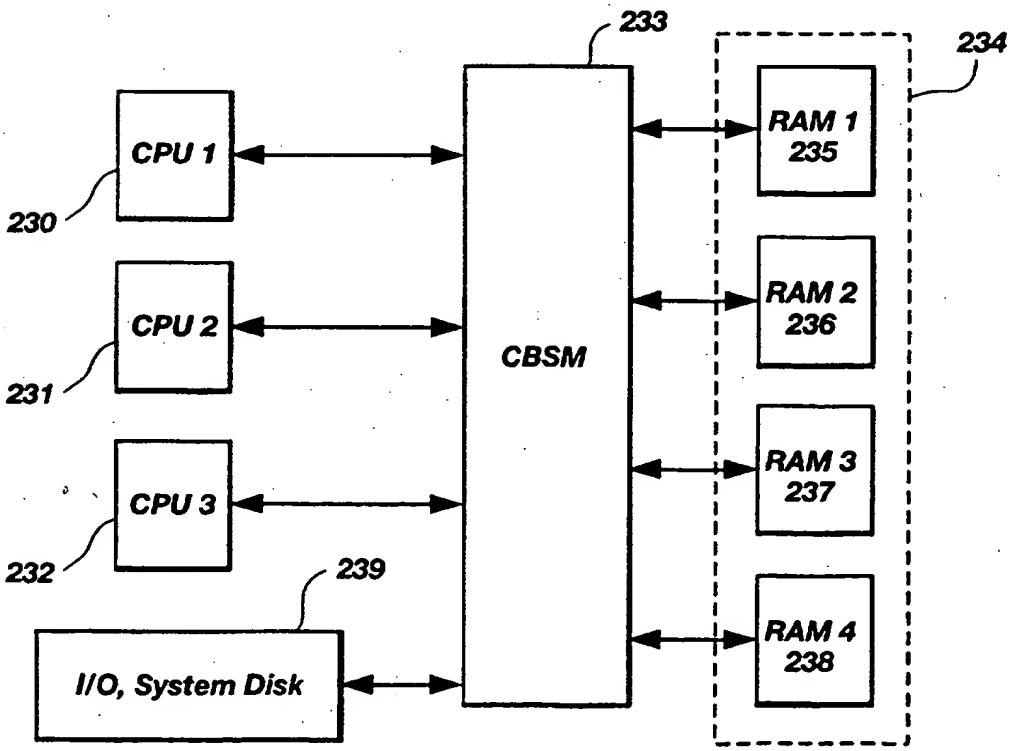


Fig. 5B

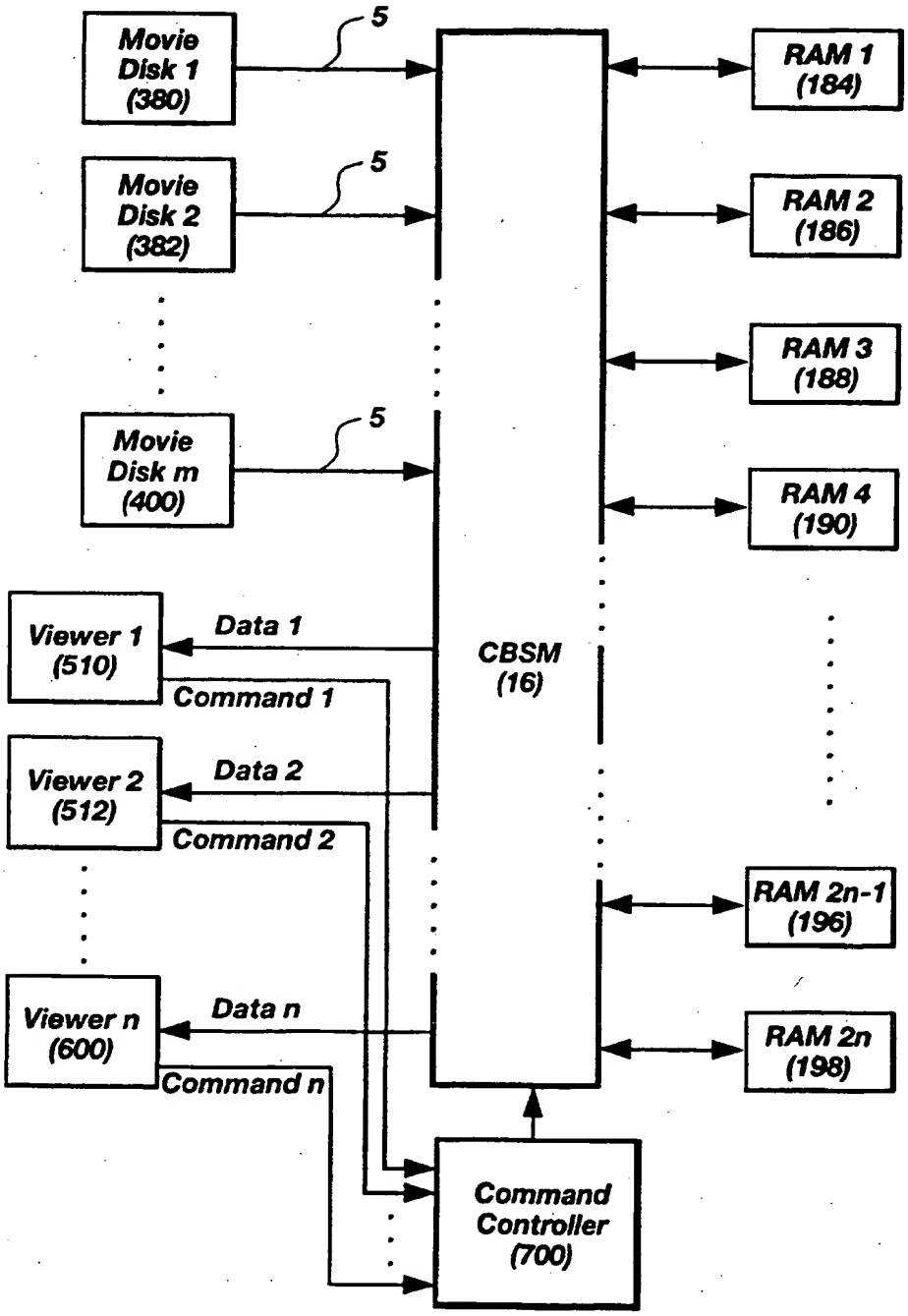

Movie Disk 1 (380)
Movie Disk 2 (382)
Movie Disk m (400)
5
5
5
RAM 1 (184)
RAM 2 (186)
RAM 3 (188)
RAM 4 (190)
RAM 2n-1 (196)
RAM 2n (198)
CBSM (16)
Viewer 1 (510)
Viewer 2 (512)
Viewer n (600)
Data 1
Data 2
Data n
Command 1
Command 2
Command n
Command Controller (700)

Fig. 6

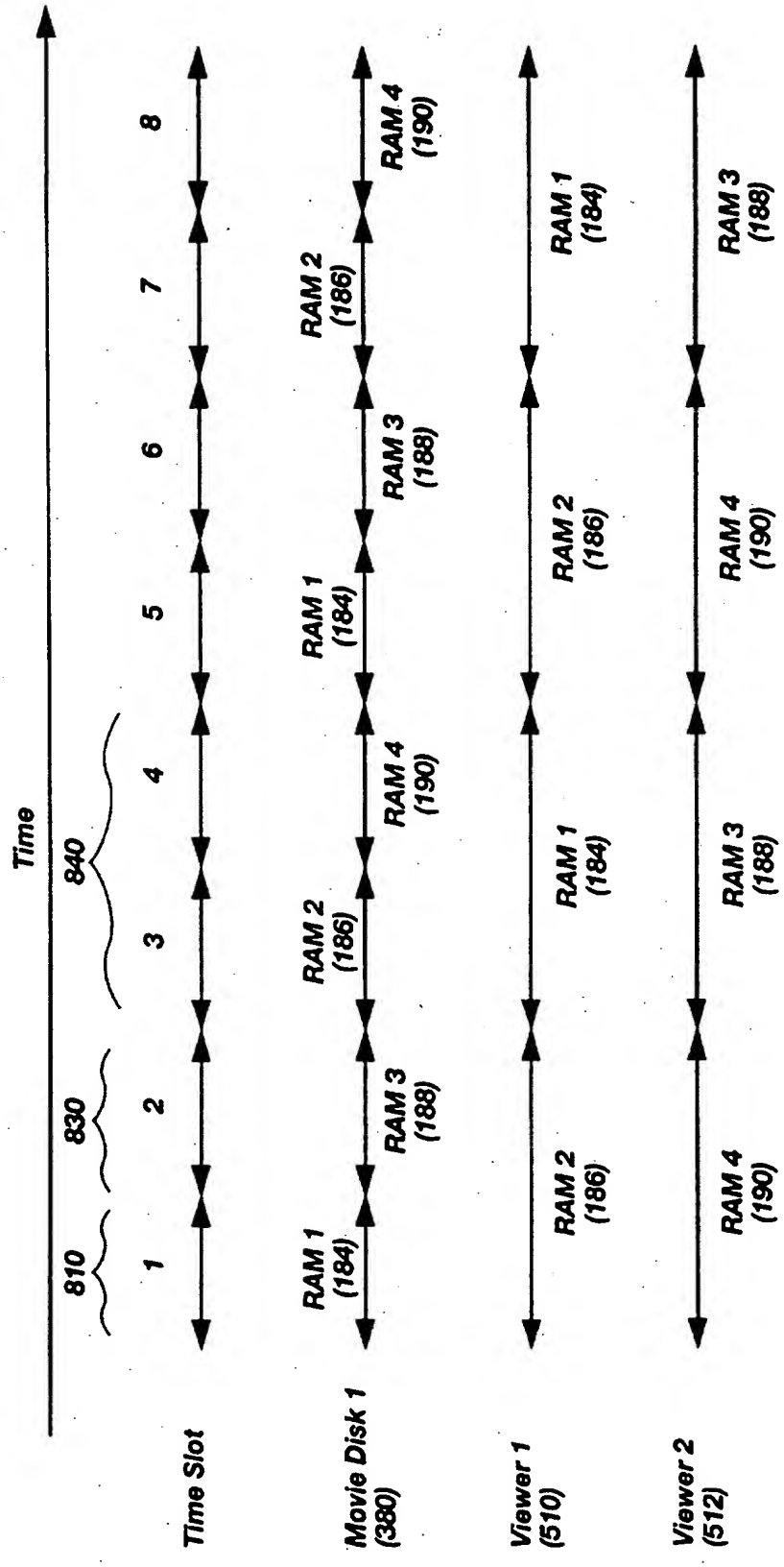
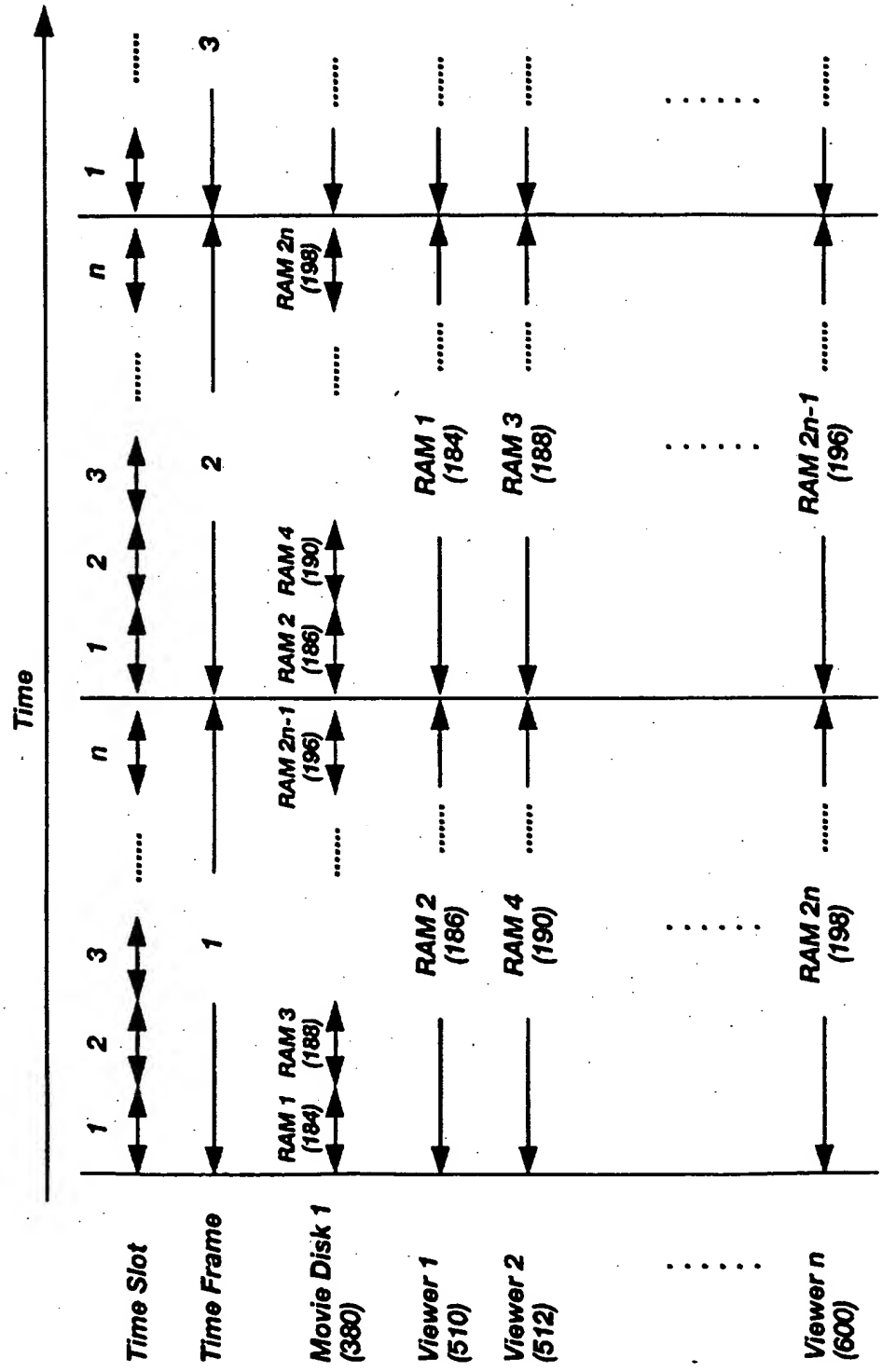

Time
Time Slot
1
2
3
4
5
6
7
8
810
830
840
Movie Disk 1 (380)
RAM 1 (184)
RAM 2 (186)
RAM 3 (188)
RAM 4 (190)
Viewer 1 (510)
Viewer 2 (512)
820

Fig. 7A



Time Slot = Time period for a Movie Disk to transfer data into a single RAM bank
Time Frame= Time period for a Viewer to access data of a single RAM bank

Fig. 7B

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US95/13309

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :G06F 13/00, 13/14
US CL :395/312, 800
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 395/312, 800

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS, IEEE Publications on ProQuest

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US, A, 5,339,396 (MURAMATSU ET AL.) 16 August 1994, figure 3.	1-32
Y	US, A, 5,355,453 (ROW ET AL.) 11 October 1994, figures 1-3.	1-32
Y	US, A, 5,226,125 (BALMER ET AL.) 06 JULY 1993, figures 1, 10, and 49.	1-32
A	US, A, 5,067,071 (SCHANIN ET AL.) 19 November 1991, figure 1.	1-32
A	US, A, 4,979,100 (MAKRIS ET AL.) 18 December 1990, figures 4, 5, and 19.	1-32

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be part of particular relevance

"E" earlier document published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
22 JANUARY 1996	14 FEB 1996
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer ALYSSA BOWLER Telephone No. (703) 305-9702

Form PCT/ISA/210 (second sheet)(July 1992)*